APPLICATION NO. 09/846.410

TITLE OF INVENTION: Multiple Data Rate Hybrid Walsh Codes for CDMA

INVENTOR~~S~~: Urbain A. von der Embse

CLAIMS

WHAT IS CLAIMED IS:

Claim 1. (cancelled)

Claim 2. (cancelled)

Claim 3. (cancelled)

Claim 4. (cancelled)

Claim **5.** (currently amended) A method for ~~the~~ generation of ~~design and implementation of fast encoders and fast decoders for Hybrid Walsh and generalized Hybrid~~ hybrid Walsh complex orthogonal codes for CDMA, ~~channelization codes for multiple data rate users over~~ said method comprising the steps: ~~a frequency band with properties~~

classify the N=2^M N-chip Walsh codes into even codes and odd codes according to their even and odd properties about their code centers for integer M,

said Walsh codes by definition are the {+1,-1} valued N orthogonal Hadamard codes re-ordered according to their sequency values where sequency is the average rate of phase changes over each N chip code length,

classify the N N-sample discrete real Fourier transform codes into even codes and odd codes and re-order said codes according to increasing frequency,

construct a one-to-one correspondence of said N real Walsh codes

with said N real Fourier transform codes such that sequency corresponds to frequency, even codes correspond to even codes, and odd codes correspond to odd codes,

construct a mapping which uses said N real Fourier codes for the real and imaginary axis codes of the N N-sample discrete complex Fourier transform (DFT) codes and,

use said mapping combined with said correspondence to generate the real and imaginary axis component codes of said hybrid Walsh codes $\tilde{W}(c)$ for code index c=0,1,2,. . .,N-1 as re-orderings of said real Walsh codes W(c) for c=1,2,. . .,N-1 defined by the equations

for $c = 0$, $\tilde{W}(c) = W(0) + jW(0)$,

for $c = 1,2,\ldots,N/2-1$, $\tilde{W}(c) = W(2c) + jW(2c-1)$,

for $c = N/2$, $\tilde{W}(c) = W(N-1) + jW(N-1))$, and

for $c = N/2+1,\ldots,N-1$, $\tilde{W}(c) = W(2N-2c-1) + jW(2N-2c)$.

~~Hybrid Walsh inphase (real axis) codes and quadrature (imaginary axis) codes are defined by lexicographic reordering permutations of the Walsh code~~

~~Hybrid Walsh codes have a 1-to-1 sequency-frequency correspondence with the DFT codes and have a 1-to-1 even-cosine and odd-sine correspondences with the DFT codes~~

~~Hybrid Walsh codes take values {1+j, -1+j, -1-j, 1-j} or equivalently take values {1, j, -1, -j} with a (-45) rotation of axes and a renormalization~~

~~generalized Hybrid Walsh codes can be constructed for a wide range of code lengths by combining Hybrid Walsh with DFT (discrete Fourier transform), Hadamard and other orthogonal~~

~~codes, and quasi-orthogonal PN codes using tensor product, direct product, and functional combining~~

~~fast encoding and fast decoding implementation algorithms are defined~~

~~algorithms are defined to map multiple data rate user data symbols onto the code input data symbol vector for fast encoding and the inverses of these algorithms are defined for recovery of the data symbols with fast decoding~~

~~encoders perform complex multiply encoding of complex data to replace the current Walsh real multiply encoding of inphase and quadrature data~~

~~decoders perform complex conjugate transpose multiply decoding of complex data to replace the current Walsh real multiply decoding of inphase and quadrature data~~

Claim **6.** (currently amended) A method for ~~the~~ generation of ~~design and implementation of encoders and decoders for complex orthogonal CDMA and~~ generalized hybrid Walsh codes for CDMA from code sets which include said hybrid Walsh, said Hadamard, said Walsh, said DFT, and pseudo-noise PN, said method comprising: ~~complex orthogonal CDMA channelization codes for multiple data rate users over a frequency band with properties~~

tensor product also called Kronecker product is used to construct said codes,

direct product is used to construct said codes,

functional combining is to construct said codes and,

combinations of tensor products, direct products, and functional combining are used to construct said codes.

~~complex codes inphase (real axis) codes and quadrature (imaginary axis) codes are defined by reordering permutations of the real Walsh codes~~

~~generalized complex codes can be constructed for a wide range of code lengths by combining the complex codes with DFT (discrete Fourier transform), Hybrid Walsh, Hadamard and other orthogonal codes, and quasi-orthogonal PN codes using tensor product, direct product, and functional combining~~

~~fast encoding and fast decoding implementation algorithms are defined~~

~~algorithms are defined to map multiple data rate user data symbols onto the code input data symbol vector for fast encoding and the inverses of these algorithms are defined for recovery of the data symbols with fast decoding~~

~~encoders perform complex multiply encoding of complex data to replace the current Walsh real multiply encoding of inphase and quadrature data~~

~~decoders perform complex conjugate transpose multiply decoding of complex data to replace the current Walsh real multiply decoding of inphase and quadrature data~~

Claim **7**. (currently amended) A method for mapping multiple data rate user symbols onto the code vectors of said codes in claime **5** and **6**, said method comprising the steps:

assign said users with like data symbol rates to the M groups

$u_{M-1}, u_{M-2}, . . ., u_1, u_0$ of users with the respective symbol rates 1/NT,2/NT,. . .,2/T in units of data symbols per second where N=2^M is the number of code chips in said code block, and number of N-chip length codes, and number of user data symbols, M is the number of said user groups and said code rates in said menu, 1/T is the code chip rate in chips per second,

in said assignment said lowest available symbol rate 1/NT user requires 1 N-chip code to support the data rate, said 2/NT user requires 2 N-chip codes, . . ., and said 2/T user requires N/2 N-chip codes,

generate the N data symbol index $d=d_0+2d_1+4d_2+...+ (N/2)d_{M-1}$ and partition said data symbol index into M fields d_{M-1}, $d_{M-2}d_{M-1}$, . . ., $d_1d_2\cdots d_{M-2}d_{M-1}$, $d_0d_1d_2\cdots d_{M-2}d_{M-1}$ which fields respectively are indexed over the available number 2,4,. . ., N/2,N of data rate users for each data symbol rate in the menu 1/2T,1/4T,. . .,1/NT symbols per second respectively,

assign said data symbol indices in field d_{M-1} to said users in said group u_{M-1}, assign said data symbol indices in field $d_{M-2}d_{M-1}$ to said users in said group u_{M-2}, . . ., assign said data symbol indices in field $d_1d_2\cdots d_{M-2}d_{M-1}$ to said users in said group u_1, and finally assign said data symbol indices in field $d_0d_1d_2\cdots d_{M-2}d_{M-1}$ to said users in said group u_1,

assign said data symbol index $d=d_0+2d_1+4d_2+...+ (N/2)d_{M-1}$ to said N-chip code vectors and,

said assigments define the mapping of said user symbols for data rates from said menu 1/NT,2/NT,. . .,2/T symbols per second onto said N-chip code vectors.

Claim **8**. (currently amended) Wherein said hybrid Walsh codes in claims **5** have a fast encoding algorithm, comprising the steps:

use said index fields in claim **7** to arrange the input data symbol set in the format $Z(d_0, d_1, . . ., d_{M-2}, d_{M-1})$ corresponding to said $d=d_0+2d_1+4d_2+...+(N/2)d_{M-1}$,

implement pass 1 of said fast encoding algorithm by multiplying said Z by the kernal $[(-1)\hat{}dr_0 n_{M-1}+j(-1)\hat{}di_0\ n_{M-1}]$ and summing over $dr_0, di_0=0,1$ to yield the partially encoded symbol set $Z(n_{M-1}, d_1, . . ., d_{M-2}, d_{M-1})$ where $dr_0=cr(d_0)$ and $cr(d)$ is the real axis code for d, $di_0=ci(d_0)$ where $ci(d)$ is the imaginary axis code for d, and n_{M-1} is a binary code chip coefficient in said code chip indexing $n = n_0+2n_1+. . .+(N/4)n_{M-2}+(N/2)n_{M-1}$,

implement passes $m=2,3,. . .,M-1$ of said fast encoding algorithm by multiplying $Z(n_{M-1}, n_{M-2}\ . . ., n_{M-m+1}, d_{m-1}, . . ., d_{M-2}, d_{M-1})$ by the kernal $[(-1)\hat{}dr_{m-1}(n_{M-m}+n_{M-m+1})+j(-1)\hat{}di_{m-1}(n_{M-m}+n_{M-m+1}])]$ and summing over $dr_{m-1}, di_{m-1}=0,1$ to yield the partially encoded symbol set $Z(n_{M-1}, n_{M-1}, n_{M-2}\ . . ., n_{M-m}, d_m, . . ., d_{M-2}, d_{M-1})$,

implement pass M of said fast encoding algorithm by by multiplying $Z(n_{M-1}, n_{M-2}\ . . ., n_2, n_1, d_{M-1})$ by the kernal $[(-1)\hat{}dr_{M-1}(n_0+n_1)+j(-1)\hat{}\ di_{M-1}(n_0+n_1)]$ and summing over $dr_{M-1}, di_{M-1}=0,1$ to yield the encoded symbol set $Z(n_{M-1}, n_{M-1}, n_{M-2}\ . . ., n_2, n_1, n_0)$, and

reorder the encoded symbol set in the ordered output format $Z(n_0, n_1, . . ., n_{M-2}, n_{M-1})$.

Claim **9**. (currently amended) Wherein said hybrid Walsh codes in claims **5** have a fast decoding algorithm, comprising the steps:

implement pass 1 of said fast decoding algorithm by multiplying said $Z(n_0, n_1, . . ., n_{M-2}, n_{M-1})$ from claim **8** by the kernal $[(-1)\hat{}n_0 dr_{M-1}+j(-1)\hat{}n_0 di_{M-1}]$ and summing over $n_0=0,1$ to yield the partially decoded symbol set $Z(d_{M-1}, n_1, . . ., n_{M-2}, n_{M-1})$,

implement passes $m=2,3,. . .,M-1$ of said fast decoding algorithn

by multiplying

$Z(d_{M-1}, d_{M-2} \ldots, d_{M-m+1}, n_{m-1}, \ldots, n_{M-2}, n_{M-1})$ by the kernal

$[(-1)\hat{}n_{m-1}(dr_{M-m} + dr_{M-m+1})+j(-1)\hat{}n_{m-1}(di_{M-m} + di_{M-m+1})]$ and summing

over $n_{m-1}=0,1$ to yield the partially decoded symbol set

$Z(d_{M-1}, d_{M-1}, d_{M-2} \ldots, d_{M-m}, n_m, \ldots, n_{M-2}, n_{M-1})$,

implement pass M of said fast decoding algorithm by

by multiplying $Z(d_{M-1}, d_{M-2} \ldots, d_2, d_1, n_{M-1})$ by the kernal

$[(-1)\hat{}n_{M-1}(dr_0 + dr_1)+j(-1)\hat{}n_{M-1}(di_0 + di_1)]$ and summing over

$n_{M-1}=0,1$ and rescaling by dividing by 2N to yield the encoded symbol set

$Z(d_{M-1}, d_{M-1}, d_{M-2} \ldots, d_2, d_1, d_0)$, and

reorder the decoded symbol set in the ordered output format

$Z(d_0, d_1, \ldots, d_{M-2}, d_{M-1})$.